

Informe de Laboratorio

Proyecto: EMPLEOYA

Nota

Estudiantes	Escuela	Asignatura
Piero De La Cruz Mancilla Jerson Ernesto Chura Pacci pdelacruz@ulasalle.edu.pe jchurap@ulasalle.edu.pe	Carrera Profesional de Ingeniería de Software	Ingeniería Web Semestre: VII Código: 20231001

Laboratorio	Tema	Duración
	Proyecto: EMPLEOYA	

1. Introducción

- **Nombre del Proyecto:** EMPLEOYA.
- **Tipo:** Aplicación Web Full-Stack.
- **Equipo:** 2 integrantes.
- **Curso:** Ingeniería Web.
- **Objetivo General:** Desarrollar una plataforma web que conecte postulantes con empleadores, facilitando la publicación de ofertas de trabajo y el proceso de postulación de manera eficiente y segura.

2. Alcance Funcional

2.1. Tipos de usuarios

- **Postulante:** Persona que busca empleo.
- **Empleador:** Empresa o reclutador que ofrece empleos.
- **Administrador:** Modera la plataforma.

2.2. Funcionalidades principales del Postulante

- Registrarse e iniciar sesión.
- Crear y editar perfil profesional.
- Subir CV en PDF.
- Buscar ofertas por título, ubicación, salario.
- Filtrar ofertas por categoría, modalidad, experiencia.
- Postular a ofertas de interés.
- Ver estado de sus postulaciones.
- Recibir notificaciones de nuevas ofertas.

2.3. Funcionalidades principales del Empleador

- Registrarse como empresa.
- Crear ofertas de trabajo detalladas.
- Ver lista de postulantes por oferta.
- Descargar CVs de postulantes.
- Marcar postulantes como “En proceso” o “Descartado”.
- Gestionar sus ofertas activas/inactivas.
- Ver estadísticas de sus publicaciones.

2.4. Funcionalidades principales del Administrador

- Moderar ofertas antes de publicación.
- Suspende usuarios problemáticos.
- Ver dashboard con métricas generales.
- Gestionar categorías de empleos.
- Generar reportes de actividad.

2.5. Límites del sistema

- No es una red social (sin posts ni conexiones).
- No procesa pagos ni suscripciones.
- No incluye video llamadas o entrevistas virtuales.
- No maneja flujo completo de RRHH.
- No tiene chat complejo (solo notificaciones del sistema).
- No hace matching automático avanzado.

3. Arquitectura Técnica

3.1. Stack Tecnológico

- **Frontend:** React 18 + Tailwind CSS + Context API.
- **Backend:** Node.js + Express.js + JWT.
- **Base de Datos:** MySQL + ORM.
- **Contenedores:** Docker + Docker Compose.
- **Storage:** Supabase (almacenamiento de CVs).
- **Control de Versiones:** Git + GitHub.

3.2. Capa de Presentación (Frontend)

- React 18 con hooks modernos para gestión de estado.
- Tailwind CSS para diseño responsivo y componentes reutilizables.
- Context API para gestión global de autenticación y datos.
- Cliente HTTP para comunicación con la API REST.
- Navegación SPA (Single Page Application).
- Responsive Design compatible con móviles y desktop.

3.3. Capa de Lógica de Negocio (Backend)

- Node.js y Express.js para manejo de rutas RESTful.
- Middleware JWT para autenticación y autorización por roles.
- Middleware de validación, CORS y manejo de errores.
- Encriptación de contraseñas y datos sensibles.
- Upload de archivos para manejo de CVs.
- Sanitización y validación de inputs del usuario.

3.4. Capa de Datos

- **MySQL** como base de datos relacional.
- Entidades principales: Usuarios, Empresas, Ofertas de trabajo, Postulaciones, Categorías.
- ORM para modelado de datos, migraciones y consultas.
- Relaciones entre usuarios, ofertas y postulaciones.

3.5. Servicios Externos y Contenedorización

- Supabase Storage para CVs en PDF con URLs seguras.
- Servicios de email o SMS según necesidad.
- Docker: contenedores individuales para frontend, backend y base de datos.
- Docker Compose para orquestación y red interna.
- Variables de entorno para configuración de servicios.
- Volúmenes para persistencia de datos.

3.6. APIs Principales

- Autenticación: registro, login y gestión de sesiones.
- Usuarios: gestión de perfiles y roles.
- Ofertas: CRUD de ofertas con filtros y búsqueda.
- Postulaciones: sistema de aplicaciones a ofertas.
- Administración: gestión de contenido y usuarios.
- **Nota:** Se pueden añadir endpoints adicionales según requerimientos específicos durante el desarrollo.

4. Entregables y Criterios de Éxito

4.1. Entregables Principales

- Código fuente completo en GitHub con README detallado.
- Aplicación web funcional deployada con contenedores.
- Base de datos implementada con datos de prueba.
- Documentación técnica completa con instalación y uso de APIs.
- Manual de usuario con guía completa de funcionalidades.

4.2. Criterios de Éxito Técnicos

- Sistema completo de autenticación y seguridad con JWT y validación.
- Control de acceso completo por roles (postulante/empleador/admin).
- Implementación completa de medidas de seguridad OWASP.
- CRUD completo para usuarios, ofertas y postulaciones.
- Sistema de búsqueda y filtros completamente operativo.
- Funcionalidad completa de upload y descarga de CVs.
- Dashboard diferenciado por tipo de usuario completamente funcional.
- Sistema de notificaciones completamente implementado.

- Interfaz responsive completa y formularios con validación completa.
- API REST completamente estructurada y validación completa en servidor.
- Base de datos con integridad completa y consultas optimizadas.
- Contenedores funcionando correctamente y deploy completo con un comando.

4.3. Métricas de Calidad

- **Performance:** Tiempos de respuesta aceptables.
- **Usabilidad:** Navegación intuitiva y eficiente.
- **Responsividad:** Compatibilidad multi-dispositivo completa.
- **Estabilidad:** Funcionamiento sin errores críticos.
- **Documentación:** Código comentado y README completo.